

REMARKS/ARGUMENTS

Favorable reconsideration of this application is respectfully requested.

Claims 1, 3, 7-9, 11, 15, 16, 18, 19, and 21-24 re pending in this application. The claims were objected to for informalities. Claims 1, 3, 7-9, 11, 15, 16, and 18 were rejected under 35 U.S.C. §112, second paragraph. Claims 1, 3, 7-9, 11, 15, 16, 18, 19, and 21-24 were rejected under 35 U.S.C. §103(a) as unpatentable over U.S. patent 5,613,120 to Palay et al. (herein "Palay") in view of U.S. patent 6,182,281 to Nackman et al. (herein "Nackman").

Addressing first the objection to the claims, the claims are amended as suggested in paragraph 1 of the Office Action to address the objections thereto.

Addressing now the rejection of Claims 1, 3, 7-9, 11, 15, 16, and 18 under 35 U.S.C. §112, second paragraph, that rejection is traversed by the present response. More particularly, each of the claims is amended by the present response to address the language noted as unclear. The claim amendments are believed to address the rejection of Claims 1, 3, 7-9, 11, 15, 16, and 18 under 35 U.S.C. §112, second paragraph.

Addressing now the rejection of Claims 1, 3, 7-9, 11, 15, 16, 18, 19, and 21-24 under 35 U.S.C. §103(a) as unpatentable over Palay in view of Nackman, that rejection is traversed by the present response.

Applicants initially note the basis for the outstanding rejection is unclear as it cites the teachings in Palay and Nackman in the statement of the rejection on the bottom of page 3 of the Office Action. However, in the body of the rejection teachings in the Background Art are also cited, particularly with respect to features recognized as neither taught nor suggested by Palay nor Nackman. Thus, it is unclear if the outstanding rejection is based solely on a combination of teachings of Palay in view of Nackman or a three-way combination based on a combination of teachings of Palay, Nackman, and certain admitted art. Clarification of the rejection if maintained is respectfully requested.

With respect to the rejection, however, applicants note the claims are amended by the present response to clarify features recited therein in a manner believed to clearly distinguish the claims over the above-noted rejection under either of the above-noted interpretations.

With respect to independent claim 1, independent claim 1 clarifies the operation of the "first source updating module" and "second source updating module". According to features clarified in independent claim 1, if a name of a detected data type definition is registered, the data type definition is deleted "from all source programs that are succeedingly compiled and linked to one object program" and the data type definition of which the use flag is "not" set is deleted from "all the source programs that are succeedingly compiled and linked to one object program". The other independent claims are similarly amended as in independent claim 1 noted above. That claim feature is believed to be clear from the original specification for example at page 15, line 26 to page 16, line 22 and Figures 13 and 14 of the present specification, as non-limiting examples.

According to features clarified in the claims as currently written, the claims address the optimization of source code (source programs) before being compiled and linked to one object program, see for example Figure 13 in the present specification, in contrast to being directed to the optimization of an object program already compiled.

According to the claimed code optimizing processor, all source programs are scanned and redundant data type definitions are deleted from all the source programs. Thereby, the optimized source programs are sent out to a compiler for compiling the source program into a relocatable object program. A plurality of relocatable object programs are succeedingly linked into one object program.

More specifically, the claimed code optimizing processor carries out a first code optimizing process, by referring to a data type definition table arranged per one object program, to delete, if the name of the detected data type definition is registered, the data type

definition from all source programs that are succeedingly compiled and linked to one object program. A second code optimizing process, by referring to a use flag in the data type definition table, deletes the data type definition of which the use flag is not set to the use status from all the source programs that are succeedingly compiled and linked to one object program to optimize the source programs.

By virtue of the claimed code optimizing processor enabling the source programs to be optimized (to be reduced in code size) before compiling and link processing, the claimed invention brings an optimized source program visible by programmers and allows the high probability of re-use of the source program, and further reduces a processing time for compiling the source programs and linking the relocatable object programs.

The above-noted features are believed to clearly distinguish over the applied art.

With respect to the claimed "first source updating module" the outstanding Office Action cites the teachings in the background section at page 7, lines 24-25.¹ However, in that respect applicants note that disclosure is unrelated to the claims as currently written. More particularly, the feature of deleting definitions for multiphase type functions whose links are duplicate in link processings is unrelated to the claimed feature of deleting the data type definition "from all source programs that are succeedingly compiled and linked to one object program".

Further in contrast to the claims as currently written, in Palay a linker removes a data type definition from an object file when the definition is already stored in the class definition table (see Palay at column 28, lines 39-61). Therefore, in Palay the subject of the code optimization is an object program already compiled from the source program. In contrast to Palay, in the claimed invention the subject of the code optimization is a source program before being compiled. As clearly recited as a problem in the background art in the

¹ Office Action of May 19, 2004, page 5, first paragraph.

specification at page 7, lines 20-28, a problem with the optimizing process of such a background system such as in Palay is that it inevitably requires a very long time for the compiling and linking processing.

As also pointed out in the Office Action the admitted art discloses a use flag variable.²

However, in that respect applicants note neither the admitted art nor Palay discloses or suggests a code optimizer for source programs that are succeedingly compiled and linked into one object program, and a data type definition table referred by the code optimizer for the source programs. As neither the admitted art nor Palay teach or suggest such features, no combination of the admitted art nor Palay discloses such a feature. Nackman also does not teach such features. Further, such a combination completely fails to disclose how to generate and update a use flag information as recited by the claimed optimizer in the claimed data type definition table.

With regard to claims 3 and 21-24, those claims are believed to be even further distinguished over the applied art. Those claims further set forth a source code optimization process for multiphase data type definition. In that respect applicants note Palay only discloses optimizing processes where only linker 112 employs a merged class instance table. In that respect, applicants submit that no matter how the admitted prior art and Palay and Nackman are combined, such a combination completely fails to disclose how to generate and update instantiation information of a multiphase data definitions recited by the claimed optimizer in the claimed data type definition table to suppress redundant instantiation in all the source programs to be succeedingly compiled and linked.

In such ways, dependent claims 3 and 21-24 are believed to even further distinguish over the applied art.

² Office Action of May 19, 2004, page 4, last paragraph.

In view of these foregoing comments, applicants respectfully submit the claims as currently written distinguish over the combination of teachings of Palay in view of Nackman, and further in view of applicants' admitted art.

As no other issues are pending in this application, it is respectfully submitted that the present application is now in condition for allowance, and it is hereby respectfully requested that this case be passed to issue.

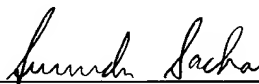
Respectfully submitted,

OBLON, SPIVAK, McCLELLAND,
MAIER & NEUSTADT, P.C.

Customer Number

22850

Tel: (703) 413-3000
Fax: (703) 413 -2220
(OSMMN 06/04)
SNS/rac



Eckhard H. Kuesters
Attorney of Record
Registration No. 28,870
Surinder Sachar
Registration No. 34,423